

REMARKS

Claims 1-4 and 6-17 were pending in the application. Claims 1, 7-9, and 13 have been amended. Accordingly, claims 1-4 and 6-17 remain pending subsequent entry of the present amendment.

35 U.S.C. § 102 Rejections

Claims 1-4 and 6-17 stand rejected under 35 U.S.C. §102(b) as being anticipated by U.S. Patent No. 5,978,791 (hereinafter “Farber”). Applicant respectfully traverses the rejections. Nevertheless, Applicant has amended the claims clarify the nature of the presently claimed invention. Accordingly, reconsideration is requested in view of the following comments.

For ease of reference, a clean version of claim 1 is reproduced below.

A method for identifying the content of a file in a network environment, said network environment comprising at least one local computing device linked to a remaining part of the network environment including a central infrastructure and, the method comprising

receiving a new file on said local computing device;
calculating a reference value for a new file using a one-way-function ;
transmitting said calculated reference value to said central infrastructure;
comparing said calculated reference value with reference values previously stored
within the remaining part of the network environment;

after comparing:

deciding that the content of the new file is already identified if a match
between said calculated reference value and a previously stored
reference value is found and retrieving corresponding content
attributes; or

deciding that the content of the new file is not yet identified if no match
between said calculated reference value and any of the previously
stored reference values is found, followed by sharing the new file
on the local computing device to said central infrastructure and
said central infrastructure identifying the content of said new file
by remotely identifying the content over the network environment,
determining content attributes corresponding with the content of
the new file and storing a copy of said content attributes;

after deciding, triggering an action on said local computing device in
accordance with said content attributes;

wherein said triggering an action on said local computing device in accordance with said content attributes comprises replacement of the new file on the local computing device with a different version of said new file restored from the remaining part of the network environment.

As seen from the above, the local computing device receives a new file and computes a reference value which is transmitted to the central infrastructure. After various activities, an action triggered on the local computing device in accordance with the content attributes includes replacement of the new file on the local computing device with a different version of said new file restored from the remaining part of the network environment. In contrast, Farber does not disclose the receipt and replacement of the file with a different version as recited.

In contrast to the presently claimed invention, Farber describes obtaining a local copy of a file using its name. In particular, Farber discloses the following:

"6. Realize True File from Location

This mechanism is used to try to make a local copy of a True File, given its True Name and the name of a source location (processor or media) that may contain the True File. This mechanism is now described with reference to FIG. 15. First, in step S272, determine whether the location specified is a processor. If it is determined that the location specified is a processor, then send a Request True File message (using the Request True File remote mechanism) to the remote processor and wait for a response (Step S274). If a negative response is received or no response is received after a timeout period, this mechanism fails. If a positive response is received, enter the True File returned in the True File registry 126 (Step S276). (If the file received was compressed, enter the True File ID in the compressed File ID field.) If, on the other hand, it is determined in step S272 that the location specified is not a processor, then, if necessary, request the user or operator to mount the indicated volume (Step S278). Then (Step S280) find the indicated file on the given volume and assimilate the file using the Assimilate Data Item primitive mechanism. If the volume does not contain a True File registry 126, search the media inventory to find the path of the file on the volume. If no such file can be found, this mechanism fails. At this point, whether or not the location is determined (in step S272) to be a processor, if desired, verify the True File (in step S282)." (Farber, col. 16, lines 10-37).

As seen from the above, Farber does not disclose the local computing device receiving a new file, calculating a reference for it which is conveyed to the central infrastructure, and having an action triggered on which causes replacement of the new file with a different version of that file. Rather, Farber discloses a device has the name of a file and wishes a copy of that file. If the file is found, a copy may be obtained. However, different versions of a file as recited aren't disclosed or contemplated. Nor is the recited replacement disclosed or suggested by Farber. For at least these reasons, claim 1 is not anticipated by Farber. Claims 7 and 9 are similarly distinguishable and are not anticipated for at least these reasons.

In addition to the above, claim 9 recites further features neither disclosed nor suggested by the cited art. For example, claim 9 includes the features

“identifying the content of said file and determining content attributes corresponding with the content of the file and storing a copy of said content attributes
-sending the content attributes to every local computing device containing the file.”

The present Office Action rejects claim 9 and states Farber discloses “determining content attributes corresponding with the content of the file and storing a copy of said content attributes sending the content attributes to every local computing device containing the file after sending (col. 23 line 53 through col. 24 line 29, col. 25 line 25-45)”. For ease of review, the cited disclosures are reproduced below:

“1. Locate True File

First determine if the True File is available locally or if there is some indication of where the True File is located (for example, in the Source IDs field). Look up the requested True Name in the True File registry 126 (Step S432). If a True File registry entry record 140 is not found for this True Name (Step S434), and the flag indicates that the request is not to be forwarded (Step S436), respond negatively (Step S438). That is, respond to the effect that the True File is not available. On the other hand, if a True File

registry entry record 140 is not found (Step S434), and the flag indicates that the request for this True File is to be forwarded (Step S436), then forward a request for this True File to some other processors in the system (Step S442). If the source table for the current processor identifies one or more publishing servers which should have a copy of this True File, then forward the request to each of those publishing servers (Step S436). If a True File registry entry record 140 is found for the required True File (Step S434), and if the entry includes a True File ID or Compressed File ID (Step S440), respond positively (Step S444). If the entry includes a True File ID then this provides the identity or disk location of the actual physical representation of the file or file segment required. If the entry include a Compressed File ID, then a compressed version of the True File may be stored instead of, or in addition to, an uncompressed version. This field provides the identity of the actual representation of the compressed version of the file. If the True File registry entry record 140 is found (Step S434) but does not include a True File ID (the File ID is absent if the actual file is not currently present at the current location) (Step S440), and if the True File registry entry record 140 includes one or more source processors, and if the request can be forwarded, then forward the request for this True File to one or more of the source processors (Step S444).” (Farber, col. 23, line 59 – col. 24, line 29).

“6. Acquire True File

This mechanism allows a remote processor to insist that a local processor make a copy of a specified True File. It is used, for example, when a cache client wants to write through a new version of a file. The Acquire True File mechanism begins with a data item and an optional True Name for the data item and proceeds as follows:

- (A) Confirm that the requesting processor has the right to require the local processor to acquire data items. If not, send a negative reply.
- (B) Make a local copy of the data item transmitted by the remote processor.
- (C) Assimilate the data item into the True File registry of the local processor.
- (D) If a True Name was provided with the file, the True Name calculation can be avoided, or the mechanism can verify that the file received matches the True Name sent.
- (E) Add an entry in the dependent processor list of the true file registry record indicating that the requesting processor depends on this copy of the given True File.
- (F) Send a positive reply.” (Farber, col. 25, lines 25-45).

The first disclosure of Farber above concerns the location of a particular file. If the particular file is not available locally (and there is an registry entry), then a request for the file may be forwarded to another processor as indicated by the registry entry record.

However, this disclosure does not disclose or suggest the features “sending the content attributes to every local computing device containing the file.”

The second disclosure of Farber above concerns one processor (a remote processor) instructing another processor (a local processor) to make a copy of a particular file. As described, the remote processor sends a file to a local processor and instructs the local processor to make a copy of the file. If the remote processor has such authority, then the local processor makes a local copy. Else, the local processor responds in the negative. Then an entry may be added to a list (the dependent processor list), and a positive reply sent. However, again, this disclosure does not disclose or suggest the features “sending the content attributes to every local computing device containing the file.” For at least the above reasons, claim 9 is patentably distinguishable from the cited art.

In light of the foregoing amendments and remarks, Applicants submit that all pending claims are now in condition for allowance, and an early notice to that effect is earnestly solicited.

If any issues remain, Applicant requests the examiner telephone the below signed representative so that an interview may be conducted.

CONCLUSION

Applicant submits the application is in condition for allowance, and an early notice to that effect is requested.

If any extension of time (under 37 C.F.R. § 1.136) is necessary to prevent the above referenced application from becoming abandoned, Applicant hereby petitions for such an extension. If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/6142-00504/RDR.

Respectfully submitted,

/ Rory D. Rankin /

Rory D. Rankin
Reg. No. 47,884
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin,
Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800

Date: June 24, 2010